ifm electronic



Manuel d'utilisation
Codeur avec interface CANopen

efector400®

FR

RM7xxx
RN7xxx

704434/01 08 / 2010 N° d'identification 685290-01

Contenu

FR

1 Remarque préliminaire

1.1 Symboles utilisés

► Action à faire
> Retour d'information, résultat
[…] Désignation d'une touche, d'un bouton ou d'un affichage
→ Référence

Remarque importante
Le non-respect peut aboutir à des dysfonctionnements ou perturbations.

Information
Remarque supplémentaire

1.2 Avertissements utilisés

ATTENTION

Avertissement de dommages matériels.

2 Consignes de sécurité

Cette notice fait partie de l'appareil. Elle fournit des textes et des figures pour l'utilisation correcte de l'appareil et doit être lue avant installation ou emploi.

Respecter les indications de cette notice.

Le non-respect des consignes, l'emploi non conforme par rapport aux prescriptions, un montage ou une manipulation incorrect peuvent porter atteinte à la sécurité des personnes et des installations.

Le montage et le raccordement doivent être conformes aux normes nationales et internationales en vigueur. La personne qui installe l'appareil en est responsable.

Seuls les signaux indiqués dans les données techniques ou imprimés sur l'appareil doivent être raccordés aux connexions.

3 Informations générales

3.1 Codeurs absolus

Avec un codeur absolu, chaque position angulaire est attribuée à une valeur de position codée. Cette valeur est générée par un disque codé avec plusieurs segments de code parallèles fins, qui sont détectés individuellement. Avec un codeur mono-tour, c'est-à-dire un codeur qui génère des positions absolues dans un tour, l'information de position absolue est répétée à chaque tour. Un codeur multi-tours peut également distinguer entre les tours via une boîte à engrenage qui contient des aimants qui sont détectés individuellement par des éléments à effet Hall. Le nombre de tours individuels est déterminé par la résolution de la détection multi-tours et est répété quand la résolution totale est atteinte.

FR

3.2 Technologie CANopen

Le profil de communication CANopen est basé sur la spécification CAN Application Layer (CAL) de CiA (CAN in Automation). CANopen est considéré comme un bus de terrain robuste avec des options de configuration très flexibles. Il est utilisé dans de nombreuses applications différentes, qui toutes reposent sur des profils d'applications différents. CANopen contient un concept pour la configuration et la communication de données de temps réel, utilisant des messages synchrones et asynchrones. On distingue quatre types de messages (objets).

1. Messages d'administration (Layer Management, Network Management et Identifier Distribution)
2. Messages Service Data (SDO)
3. Messages Process Data (PDO)
4. Messages prédéfinis (Synchronisation, Time Stamp, Emergency)

Pour plus d'informations, veuillez vous référer à la spécification CANopen.

3.2.1 Certification des produits CANopen

Afin d'atteindre l'interopérabilité et une fonctionnalité d'appareil appropriée, les produits CANopen sont vérifiés par des organismes de certification externes. Une copie du certificat est jointe à ce manuel.

3.3 Références

http://www.can-cia.org

CAN Application Layer, DS 201…207	CiA
Profil de communication basé sur CAL, DS 301	CiA
Profils d'appareils pour codeurs, DS 406	CiA
Spécification CAN version 2.0 A	Robert Bosch GmbH
Commande CAN CANary	Atmel

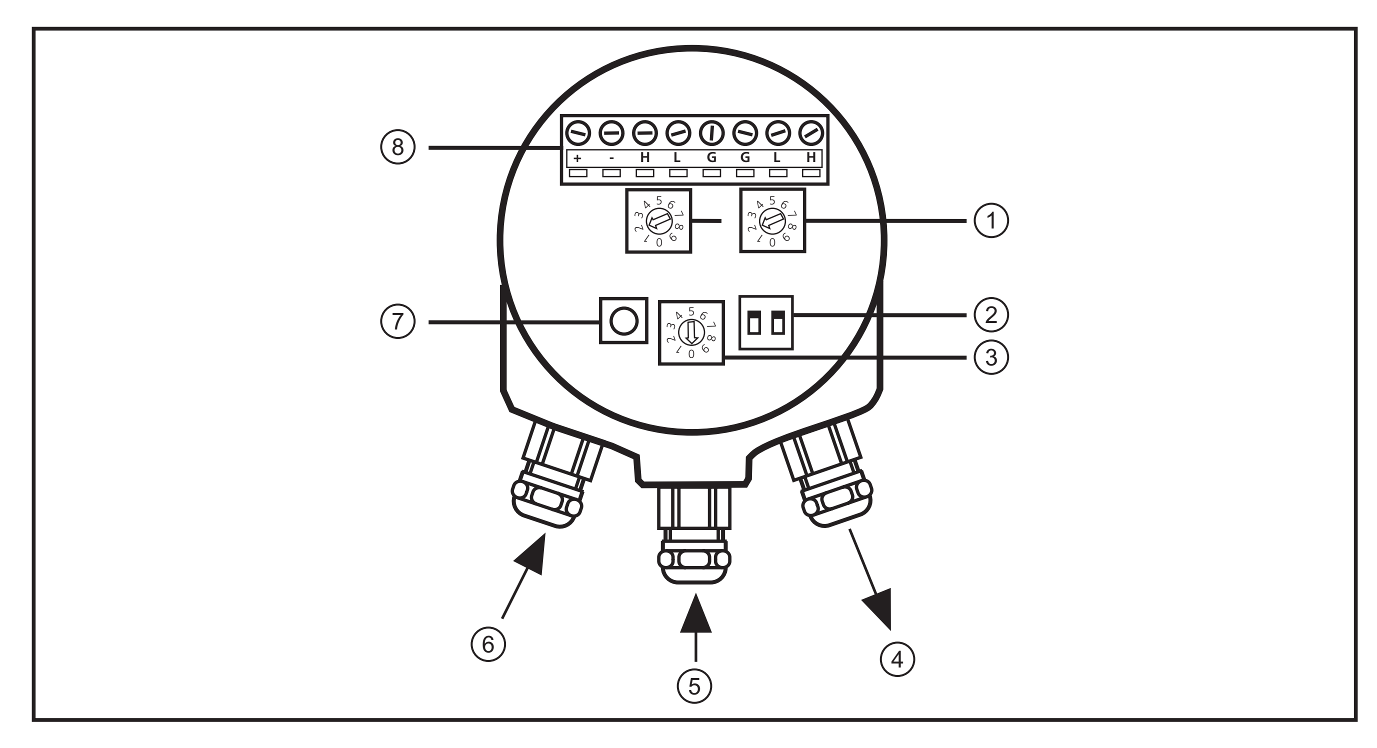
3.3.1 Abréviations

CAN	Controller Area Network
CiA	CAN in Automation
CAL	CAN Application Layer
EDS	Electronic Data Sheet
DCF	Device Configuration File
SDO	Service Data Object
PDO	Process Data Object
TPDO	Transmit PDO
COB-ID	Communication Object Identifier
NMT	Network Management
IRT	Isochronous Real Time

4 Installation du codeur

4.1 Réglages dans le codeur

L'adresse du nœud du codeur, le débit de transmission et la terminaison de bus doivent être configurés pendant la mise en service de l'appareil. Pour cela, enlever le couvercle du boîtier sur le dos.



1: commutateur adresse du nœud
2: terminaison de bus activée/désactivée
3: commutateur de débit de transmission
4: sortie bus
5: entrée bus

6: alimentation $+U_B$
7: bouton de mise à zéro
8: bornes à vis pour le bus et le raccordement d'alimentation

4.2 Adresse du nœud

L'adresse du nœud de l'appareil peut être réglée via deux commutateurs rotatifs décimaux dans l'appareil. L'incrément, x10 et x1, est indiqué à côté des commutateurs. La plage d'adressage admissible est de 3 à 98. La plage d'adressage 0 à 2 est réservée au maître. L'adresse 0 est utilisée pour des messages collectifs, c'est-à-dire les messages collectifs du maître aux esclaves multiples. A noter que chaque adresse utilisée dans un réseau CANopen doit être unique et ne doit pas être utilisée par d'autres appareils.

FR

L'adresse de l'appareil est lue et adoptée quand l'alimentation du codeur est activée (ou par la commande NMT Reset_Communication ou Reset_Node). Les deux actions sont nécessaires pour adopter les modifications effectuées sur l'adressage.

4.3 Terminaison de bus

Tous les appareils dans un réseau CANopen sont liés par une structure de bus. 32 appareils (maîtres ou esclaves) au maximum peuvent être raccordés dans une section. Quand plus d'appareils sont nécessaires, des répéteurs doivent être utilisés pour amplifier les signaux entre les segments. Au début et à la fin de chaque segment de bus, une terminaison active doit être ajoutée pour assurer un fonctionnement correct. Ces terminaisons sont installées dans l'appareil et peuvent être activées via des sélecteurs DIP.

La terminaison active est seulement activée si le codeur est actif. Quand l'appareil est sans courant, les fils CAN_H- et CAN_L- sont terminés en interne par une résistance 121Ω.

Bit 1	Bit 2	Effet
actif	actif	résistance 121 ohms entre CAN_H et CAN_L
actif	inactif	aucun réglage valable
inactif	actif	aucun réglage valable
inactif	inactif	aucune résistance entre CAN_H et CAN_L

4.4 Commutateur de débit de transmission

Le débit de transmission de communication peut être réglé à l'aide du commutateur rotatif dans le codeur. Le débit de transmission est réglé selon le tableau suivant.

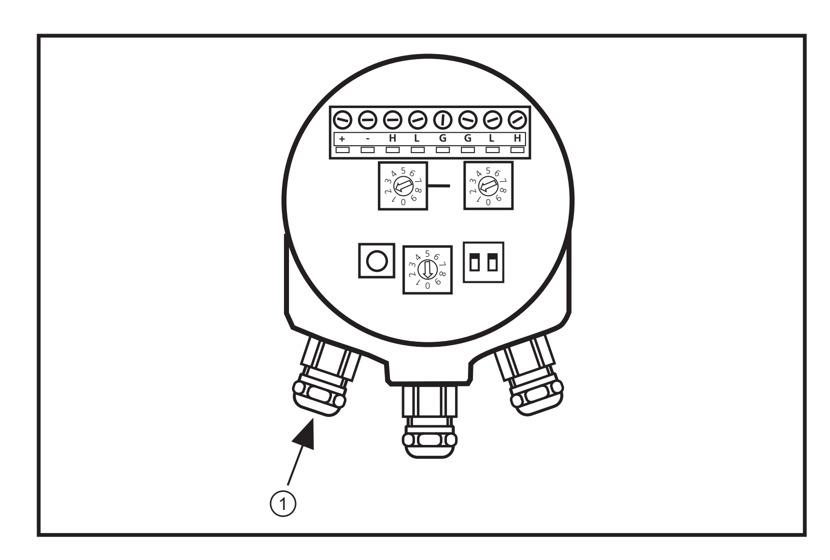
Réglages commutateur de débit de transmission

Taux de transmission [Kbit]	Commutateur de débit de transmission
10	0
20	1
50	2
125	3
250	4
500	5
800	6
1000	7
400	8

4.5 Raccordement électrique du codeur

ATTENTION

L'appareil doit être monté par un électricien qualifié.
Mettre l'installation hors tension avant de raccorder l'appareil.



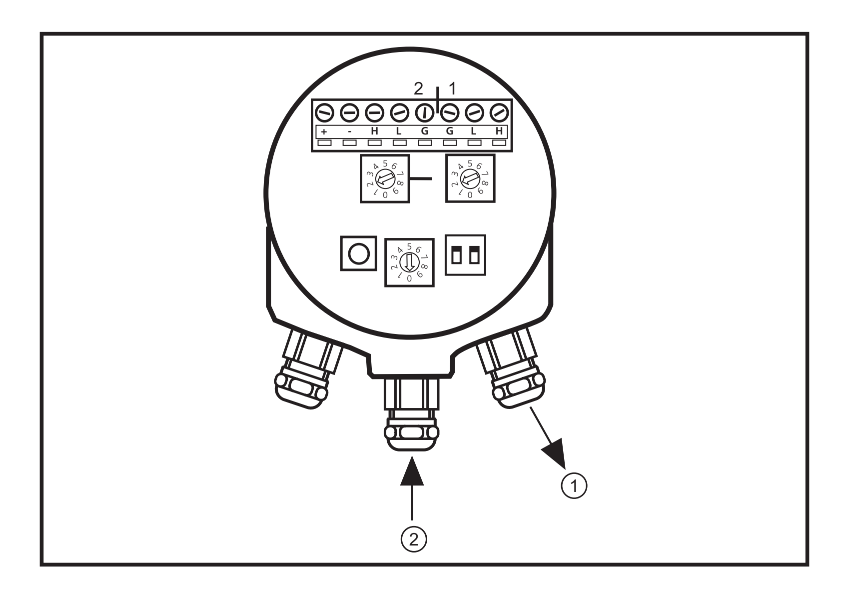
1: Tension d'alimentation U_B

Fonction	Borne
$+U_B$ (9...36 V DC)	+
0 V	-

Les entrées de câble du codeur doivent toujours être équipés d'un câble d'alimentation en tension blindé avec une section transversale des fils entre 0,34 mm^2 et 1,5 mm^2. Le diamètre extérieur admissible du câble est de ø 8...ø 10 mm. Dans le couvercle, il y a deux bornes à vis avec les bornes d'alimentation en tension nécessaires avec le marquage (+) et (-).

La borne (+) est utilisée pour le raccordement au faisceau $+U_B$ (9...36 V DC). La borne (-) est utilisée pour le raccordement au faisceau 0 V.

4.6 Faisceaux de BUS



1: sortie bus
2: entrée bus

Fonction	Borne
Blindage CAN	Entrée de câble
CAN GND	G
CAN_H	H
CAN_L	L

Les entrées de câble du codeur doivent être équipées d'un câble consistant en deux fils conducteurs individuels torsadés selon EN50170. Les directives recommandent une section transversale des fils > 0,34 mm ^{2}. Le diamètre extérieur admissible du câble est de ø 6 ... ø 8 mm. Dans la boîte de raccordement, il y a six bornes à vis avec les bornes de câble bus nécessaires marquées H, L et G.

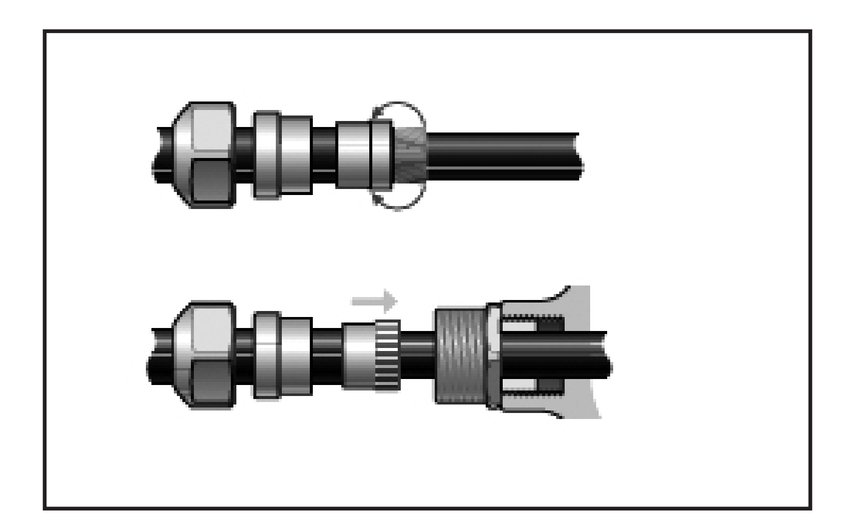
► Raccorder la borne (H) au câble CAN_H.

► Raccorder la borne (L) au câble CAN_L.

► Raccorder la borne (G) au câble CAN_G.

Les bornes G, H et L sont reliées en interne, ainsi que les câbles bus peuvent être raccordés à un pair quelconque.

4.7 Blindage



Afin d'atteindre la meilleure protection contre les tensions parasites et la meilleure compatibilité par rapport à autres parasites électromagnétiques, les câbles d'alimentation en courant et les câbles bus doivent toujours être blindés. Le blindage doit être relié à la terre sur les deux extrémités du câble. Dans certains cas, un courant de compensation peut écouler à travers le blindage. Pour cette raison, un fil de compensation est recommandé pour le potentiel.

4.8 Fichier EDS

Un fichier EDS peut être téléchargé de notre site web :

→ 

Contenu du fichier EDS :

- Les fonctions de communication et les objets, comme définis dans le profil de communication CANopen DS-301
- Les objets spécifiques à l'appareil, comme défini dans le profil de codeur DS-406
- Objets spécifiques au fabricant.

Le fichier EDS sert de modèle pour les différentes configurations d'un type d'appareil. Un fichier DCF est généré par le fichier EDS et décrit une configuration spécifique de l'appareil, y inclus des valeurs de l'objet, le débit de transmission sélectionné et l'ID de module.

Des outils de configuration CANopen sont disponibles afin de supporter la configuration de réseau CANopen et la configuration d'appareil via le bus CAN. Les informations concernant l'appareil se trouvent dans le fichier EDS.

Le processus d'installation EDS dépend de votre outil de configuration ; en cas de problèmes veuillez contacter le fabricant de votre commande.

4.9 Paramétrage

Si l'appareil se trouve à l'état Pre-Operational, les paramètres sont réglés à l'aide des objets contenus dans le fichier EDS par l'outil de configuration. Les paramètres peuvent également être modifiés pendant l'état de fonctionnement.

Les données de position sont directement influencées par certains paramètres et changent directement après un tel message de paramètre.

► Ne modifier les paramètres de la fonction de mise à l'échelle et la séquence de code que si l'arbre est à l'arrêt.

Le processus de paramétrage dépend de votre outil de configuration ; en cas de problèmes veuillez contacter le fabricant de votre commande.

FR

4.10 Affichage LED

Deux LED sont visibles sur le couvercle du codeur ; elles indiquent l'état du codeur. La LED module indique l'état du module. La LED état indique l'état du module sur le bus.

Les LED peuvent être activées ou désactivées en permanence, clignoter ou scintiller.

Clignotement : LED activée 200 ms, désactivée 200 ms

Scintillement : LED activée 200 ms, désactivée 1000 ms (bref scintillement une fois)

LED activée 200 ms, désactivée 200 ms, activée 200 ms, désactivée 1000 ms (bref scintillement deux fois)

4.10.1 LED module

La LED module est une LED bicolore avec la fonctionnalité suivante :

LED	Affichage
éteinte	aucune alimentation en tension
verte	alimentation en tension OK
rouge	défaut de position, le codeur ne peut pas indiquer de valeur de position incorrecte
3 x clignotement vert, éteinte	bouton de réglage zéro appuyé et position mise à zéro ; après 3x clignotement, la LED retourne automatiquement à l'état précédent
rouge, clignotante	réglages de commutation incorrects

4.10.2 LED d'état

La LED d'état est une LED bicolore avec deux fonctions ; une LED verte (état Run) et une LED rouge (état d'erreur).

LED	Affichage
verte, scintillante	le codeur est à l'état NMT Operational
verte, clignotante	le codeur est à l'état NMT Stopped
verte allumée	le codeur est à l'état NMT Pre-Operational
rouge éteinte	aucun défaut
rouge, clignotante	débordement du compteur d'erreurs
double clignotement rouge	un événement Guard ou Heartbeat s'est produit
rouge allumée	le codeur est à l'état Bus Off

La LED module verte est allumée et la LED d'état scintille en vert quand le codeur se trouve à l'état Operational en cas de communication correcte.

5 Aperçu de profil

Le profil de codeur définit les fonctions des codeurs raccordés à CANopen. Les fonctions de fonctionnement sont divisées en deux classes d'appareils :

Classe 1

La classe obligatoire avec des fonctions de base qui doivent être supportées par tous les codeurs. En option, le codeur de la classe 1 peut supporter des fonctions sélectionnées de la classe 2. Par contre, ces fonctions doivent être réalisées de façon correspondante au profil.

Classe 2

Le codeur doit supporter toutes les fonctions de la classe 1 ainsi que toutes les fonctions définies dans la classe 2.

Fonctions de la classe 2

- Transmission de la valeur de position absolue avec les modes scrutation, cyclique ou SYNC.
- Valeurs fournies vitesse et accélération
- Changement de la séquence de code
- Préréglages
- Mise à l'échelle de la résolution du codeur

Diagnostic avancé

- Identification du codeur
- Etat de fonctionnement
- Temps de fonctionnement
- Messages d'alarme et avertissements

Les SDO permettent l'accès à tous les paramètres de programmation et de diagnostic. La valeur de position fournie par le codeur est tout ou rien.

6 Fonctionnalité de codeur

6.1 Fonctionnalité principale de codeur

L'illustration suivante donne un aperçu des fonctions fondamentales du codeur et de la réalisation de leur fonctionnalité dans le codeur.

Position physique	
Fonction de base	← séquence de code
	→ résolution mono-tour → nombre de tours distinguables
Position absolue	
Fonction de mise à l'échelle	↔ points par tour ↔ plage de mesure complète en points ↔ fonction de mise à l'échelle commande/état
Fonction de présélection	← valeur présélectionnée
	→ valeur offset
Sortie valeur de position	

6.2 Identificateurs de défaut

Pour réduire la complexité de configuration, un aperçu d'allocation des identificateurs de défaut est défini pour les appareils CANopen. Cet aperçu d'allocation ID consiste en une partie fonctionnelle qui détermine la priorité des objets, et une partie ID de module, qui est égale au numéro de nœud (1 à 127). Un message collectif de services non confirmés (NMT et SYNC) est indiqué par une ID de module de zéro.

Dans CANopen, l'identificateur à 11 bits est structuré comme suit

Bits n° 0...6   N° de nœud

Bits n° 7...10 Code de la fonction

N° de bit	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	Code de fonction				N° de nœud						

Les objets broadcast suivants avec identificateurs de défaut sont disponibles dans le codeur :

Objet	**Code de fonction (tout ou rien)**	**Identificateur y résultant (COB-ID)**	**Groupe de priorité**
NMT	0000	0	0
SYNC	0001	128	0

Les objets poste-à-poste (peer-to-peer) suivants avec identificateurs de défaut sont disponibles dans le codeur :

Objet	**Code de fonction (tout ou rien)**	**Identificateur résultant (COB-ID)**	**Groupe de priorité**
EMERGENCY	0001	129...255	0, 1
PDO1 (tx)	0011	385...511	1, 2
PDO2 (tx)	0101	641...767	2, 3
SDO (tx)	1011	1409...1535	6
SDO (rx)	1100	1537...1663	6, 7
Node Guard	1110	1793...1919	-

6.3 Message boot

Le codeur transmet un message boot après l'activation et l'initialisation. Ce message utilise le Default Emergency Identifier et n'a pas d'octets de données. Avec ce message, l'utilisateur peut directement obtenir le nœud de transmission de l'identificateur utilisé (COB-ID), parce que c'est une fonction du numéro de nœud, → chapitre 6.2.

6.4 Paramètres de fonctionnement

Objet 6000h, paramètres de fonctionnement, commande les fonctions pour la séquence de code et la mise à l'échelle.

Bit	Fonction	Bit = 0	Bit = 1	Classe 1	Classe 2
0	séquence de code	CW	CCW	M	M
2	contrôle de la fonction de mise à l'échelle	désactivé	validé	O	M
4...11	réservé pour autre utilisation				
12...15	paramètres spécifiques au fabricant	pas d'information	pas d'information	O	O

FR

La séquence de code définit si des valeurs de position montantes ou descendantes sont fournies quand l'arbre du codeur tourne en sens horaire ou en sens anti-horaire (vue sur l'arbre). Le contrôle de la fonction de mise à l'échelle est utilisé pour la validation / le blocage des points par tour dans l'objet (6001h) et pour la plage de mesure complète en points dans l'objet (6001h) (→ chapitre 6.5.)

Quand le bit de la fonction de mise à l'échelle est mis, les paramètres de mise à l'échelle influencent la valeur de position fournie. Quand le bit de la fonction de mise à l'échelle est mis à 0, la fonction de mise à l'échelle est désactivée.

6.5 Fonction de mise à l'échelle

6.5.1 Aperçu

Avec la fonction de mise à l'échelle, la valeur numérique interne du codeur est convertie dans le logiciel, pour modifier la résolution physique du codeur. Les paramètres " points par tour " (objet 0x6001h) et " plage de mesure complète en points " (objet 0x6002h) sont les paramètres de mise à l'échelle qui fonctionnent avec le bit de contrôle de la fonction de mise à l'échelle.



Quand un codeur multi-tours est mis à l'échelle, le paramètre " points par tour " doit être transmis avant le paramètre " plage de mesure complète en points ".

Le type de données pour les deux paramètres de mise à l'échelle est 32 (sans signe) avec une plage de valeurs de 1 à 2^{32} (limitée par la résolution du codeur). Pour un codeur à 25 bits avec une résolution mono-tour de 13 bits, la valeur admissible pour " points par tour " est entre 1 et 2^{13} (8192) et pour " plage de mesure complète en points " la valeur admissible est entre 1 et 2^{25} (33554432).

Les paramètres de mise à l'échelle sont sauvegardés dans une mémoire protégée contre les coupures de tension et sont chargés de nouveau avec chaque démarrage.

Format des paramètres de mise à l'échelle mono-tour

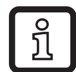
Octet	**3**	**2**	**1**	**0**
Bit	31 - 24	23 - 16	15 - 8	7 - 0
Données	2^{31} - 2^{24}	2^{23} - 2^{16}	2^{15} - 2^{8}	2^{7} - 2^{0}
Objet 6001h - points par tour				

Format des paramètres de mise à l'échelle multi-tours

Octet	**3**	**2**	**1**	**0**
Bit	31 - 24	23 - 16	15 - 8	7 - 0
Données	2^{31} - 2^{24}	2^{23} - 2^{16}	2^{15} - 2^{8}	2^{7} - 2^{0}
Objet 6002h - plage de mesure complète en points				

6.5.2 Formules de mise à l'échelle

La fonction de mise à l'échelle utilisée dans le codeur CANopen est limitéé à une résolution mono-tour en un point. Après le téléchargement de nouveaux paramètres de mise à l'échelle, la fonction de présélection devrait être utilisée pour régler le point de départ du codeur.



L'arbre doit être à l'arrêt pour modifier les paramètres de la fonction de mise à l'échelle.

Comme exemple, un codeur multi-tours à 25 bits est utilisé dans la formule suivante avec une résolution mono-tour de 13 bits.

Formule pour la fonction de mise à l'échelle multi-tours :

A = (singleturn_position x measuring_units_per_revolution) / 8192

output_position = (revolution_number x measuring_units_per_revolution) + A

Avec : singleturn_position = valeur de position mono-tour absolue

revolution_number = nombre absolu multi-tours

6.6 Valeur présélectionnée

6.6.1 Aperçu

La fonction de présélection (objet 0x6003h) supporte l'adaptation du codeur au point de zéro mécanique ou à une valeur à prérégler. La fonction de présélection est utilisée après la fonction de mise à l'échelle. La valeur préréglée est fournie comme valeur mesurée.

Une présélection est déterminée par le codeur comme suit :

Le codeur lit la valeur de position actuelle et calcule une valeur offset de la valeur présélectionnée et de la valeur de position lue. La valeur de position est déplacée par la valeur offset calculée. La valeur offset peut être lue par la fonction de diagnostic (objet 6509h), est sauvegardée dans une mémoire protégée contre les coupures de tension et est chargée de nouveau à chaque démarrage.

FR



Seulement utiliser la fonction de présélection si l'arbre est à l'arrêt.

Format de la valeur présélectionnée

Octet	**3**	**2**	**1**	**0**
Bit	31 - 24	23 - 16	15 - 8	7 - 0
Données	2^{31} - 2^{24}	2^{23} - 2^{16}	2^{15} - 2^{8}	2^{7} - 2^{0}
Objet 6003h - valeur présélectionnée				

6.6.2 Calcul de présélection

Une valeur offset est calculée quand le codeur reçoit la valeur présélectionnée, voir le calcul de setup ci-dessous. La valeur offset est ensuite utilisée pendant la durée d'exécution pour déplacer la position actuelle à la position fournie nécessaire, voir le calcul de la durée d'exécution ci-dessous.



Dans les formules ci-dessous, la position actuelle est la position absolue du disque du codeur après la fonction de mise à l'échelle. Les calculs s'effectuent avec des valeurs avec signe.

Calcul setup : *offset_value = preset_value - current_value*



Une valeur offset réglée auparavant n'est pas contenue dans la position actuelle.

Calcul de la durée d'exécution : *output_position = current_position + offset_value*

6.7 Réglage zéro

Le réglage zéro peut être effectué par 2 méthodes.

Réglage par logiciel

On reçoit un réglage zéro du codeur si l'on utilise l'objet préréglé et met la valeur préréglée à zéro (00 00 00 00h).

Réglage par bouton

La position du codeur est mise à zéro (00 00 00 00h) quand le bouton de réglage zéro est appuyé pendant au moins 1 seconde.

Affichage LED module
verte, éteinte, verte, éteinte, verte, éteinte pour confirmer la mise à zéro de la valeur de position.

6.8 Vitesse et accélération

Le codeur supporte la mise à disposition de l'objet de vitesse (0x6030) ainsi que de l'objet d'accélération (0x6040). Différents points peuvent être réglés pour retenir l'exactitude indépendamment de la vitesse de rotation du codeur. L'objet de vitesse est limité à une valeur de 16 bits avec signe. Une optimisation de la vitesse de rotation assumée de l'arbre concernant la résolution sélectionnée est nécessaire pour éviter un débordement de données.

L'objet 0x5003 (le type de vitesse) est un objet spécifique au fabricant, réglant le temps de mise à jour et la résolution (points / seconde ou tr / min) de l'objet de vitesse (0x6030) et de l'objet d'accélération. L'objet type de vitesse est décrit dans le chapitre 7.1.

6.9 Mapping PDO

Le mapping PDO dynamique permet des changements aux objets transmis à un PDO. Les codeurs des séries RM et RN peuvent mapper trois objets différents dans les PDO.

Nom	Objet	Sous-indice	Longueur
Position	0x6004		4 octets
Vitesse	0x6030	1	2 octets
Accélération	0x6040	1	2 octets

Le codeur a deux PDO à transmettre. PDO1 (transmis de manière cyclique par Cyclic_Timer) et PDO2 (transmis quand un message SYNC a été reçu). Les deux PDO sont mappés par défaut, pour transmettre uniquement des données de position. Les deux PDO peuvent être modifiés de manière indépendante pour transmettre une combinaison et une séquence de l'objet ci-dessus.

La structure des saisies de l'objet " transmission du paramètre mapping PDO " sous-indice 1...3 est comme suit :

Octet MSB	Octet MSB - 1	Octet LSB + 1	Octet LSB
Objet		Sous-indice	Longueur d'objet (nombre de bits)

6.9.1 Configuration PDO

Afin de modifier le mapping PDO, le codeur doit être à l'état Pre-Operational en mode NMT. Le PDO doit être mis à " non valable ". Cela se fait en annulant le bit 31 (MSB) dans le sous-indice 1 " COB-ID utilisé par PDO " dans l'objet " transmission des paramètres de communication PDO ".

Le PDO doit être désactivé, mettre " transmission des paramètres mapping PDO " sous-indice 0 à 0.

Pour réconfigurer le mapping PDO, transmettre les données de l'objet concerné, le sous-indice et la longueur du premier objet à " transmission des paramètres mapping PDO ", sous-indice 1. Ensuite, procéder de même pour le deuxième ou troisième objet optionnel sur " transmission des paramètres mapping PDO ", sous-indices 2 et 3.

" Transmission des paramètres mapping PDO " sous-indice 0 doit être mis au nombre des objets mappés dans le PDO (1-3).

Le mapping PDO réconfiguré doit être mis à " valable " par le bit de réglage 31 (MSB) dans le sous-indice 1 " COB-ID utilisé par le PDO " dans l'objet " transmission des paramètres de communication PDO ". Après le réglage du codeur dans le mode NMT Operational, le mapping PDO réconfiguré est activé.

Le mapping PDO peut être sauvegardé de manière sécurisée sur EEPROM par l'objet 0x1010 " sauvegarder champ de paramètres " (sous-indice 1, " tous les paramètres " ou sous-indice 2 " paramètres de communication ").

6.9.2 Exemple de configuration PDO

Le chapitre suivant montre comment le PDO1 est mappé avec la position et la vitesse (le codeur à l'adresse 0x0F, toutes les données au format Hex).

Point	ID	Données	Remarque
1	0	80 0F	Réglage du codeur à Pre-Operational en mode NMT
2	60F	23 00 18 01 8F 01 00 80	Régler PDO1 à non valable et COB-ID à 0x18F
3	60F	2F 00 1A 00 00 00 00 00	Régler " transmission des paramètres mapping PDO " sous-indice 0 à 0 (mapping désactivé)
4	60F	23 00 1A 01 20 00 04 60	Mapper la position (objet 0x6004) sur la première position dans le PDO
5	60F	23 00 1A 02 10 01 30 60	Mapper la vitesse (objet 0x6030) sur la deuxième position dans le PDO
6	60F	2F 00 1A 00 02 00 00 00	Mettre " transmission des paramètres mapping PDO " sous-indice 0 à 2 (le nombre d'objets mappés dans le PDO)
7	60F	23 00 18 01 8F 01 00 00	Régler PDO1 à valable et COB-ID à 0x18F
8	0	01 0F	Régler le codeur quand il est en mode NMT Operational

Le mapping de PDO1 est donc terminé. Le message PDO1 peut par exemple être comme suit :

ID	Données
18F	4E C9 B2 00 53 01

" 4E C9 B2 00 " sont les données de position et " 53 01 " la valeur de vitesse.

Pour sauvegarder le mapping PDO dans l'EEPROM, transmettre :

ID	Données	Remarque
60F	23 10 10 02 73 61 76 65	Sauvegarder tous les paramètres de communication en transmettant le code ASCII pour " SAUVEGARDER " à l'objet 0x1010, sous-indice 2

6.10 Heartbeat

Les codeurs des séries RM et RN peuvent agir comme " Heartbeat Producer ". Le temps entre les deux heartbeat est configuré dans l'objet " Producer Heartbeat Time " (0x1017) et il est dans la plage de millisecondes (1...65535). Quand le temps " Producer Heartbeat Time " (0x1017) est zéro (0), heartbeat est désactivé.

FR

L'objet " Heartbeat Time " (0x1017) est sauvegardé de manière sécurisée dans l'EEPROM et chargé de nouveau au démarrage.

6.11 Mode IRT

Pour améliorer la caractéristique de temps réel, le codeur peut opérer en mode IRT (Isochronous Real Time = temps réel isochrone). En mode de fonctionnement normal, la valeur de position est enregistrée de manière cyclique toutes les 0,5 ms. Si " lire la position à Sync " est bloqué, le PDO2 (transmettre données à SYNC) utilise la dernière position enregistrée du codeur. Ainsi, une caractéristique de temps non réel est ajoutée à la valeur de position fournie. En mode IRT, " lire la position à Sync " est mis, la valeur de position est seulement enregistrée quand le message SYNC est reçu.

Les modifications suivantes sont effectuées quand le bit " lire la position lors de Sync " est mis dans l'objet de paramètre de fonctionnement (0x6000) :

- L'objet de vitesse (0x6030) et l'objet d'accélération (0x6040) sont bloqués, parce que l'enregistrement cyclique de la position est obligatoire pour calculer ces valeurs.
- PDO1 (transmission cyclique de données) est bloqué, l'objet 0x1800, sous-indice 1, bit 31 est mis et sauvegardé dans l'EEPROM.
- L'objet " transmettre PDO1 " (0x1800) est seulement lu si le bit " lire la position à SYNC " est mis.
- Le PDO2 (transmettre les données à SYNC) est seulement mis pour transmettre des données de position et le nouveau mapping PDO2 est sauvegardé dans l'EEPROM.
- L'objet "PDO2 tx-Mapping" (0x1A01)" est seulement lu si le bit " lire la position à Sync " est mis.



Quand les valeurs de vitesse et d'accélération sont nécessitées pendant le fonctionnement en mode IRT, il est recommandé de calculer ces valeurs dans l'application maître et d'utiliser l'horloge maître (message SYNC) en tant que référence.

Bit	Paramètre
0	séquence de code
1	pas d'information
2	contrôle de la fonction de mise à l'échelle
3...14	pas d'information
15	lecture de la position à SYNC

Paramètres de fonctionnement (objet 0x6000)

6.11.1 Diagnostic de codeur

Le diagnostic de codeur peut être lu par les objets 65xxh. Les diagnostics d'état de fonctionnement, d'alarme et d'avertissement sont décrits dans les chapitres suivants. Pour un aperçu complet du diagnostic supporté voir le fichier EDS.

6.11.2 Etat de fonctionnement

L'état de fonctionnement peut être lu dans l'objet 6500h. La fonction pour chaque bit correspond aux paramètres de fonctionnement, voir chapitre 6.4.

Le contrôle de la fonction de mise à l'échelle (bit 2) est mis à l'état de fonctionnement en fonction du réglage dans les paramètres de fonctionnement. De plus, les valeurs réelles de mise à l'échelle utilisées dans le codeur peuvent être lues en tant que diagnostic, objet 6501h (résolution mono-tour) et objet 6502h (résolution multi-tours).

6.12 Messages d'alarme et d'avertissement

Quand le codeur enregistre une alarme interne, il passe automatiquement à l'état Pre-Operational. Le codeur envoie un message COB-ID EMCY (objet 0x1014h). Ce message transmet le type d'alarme qui s'est produit. Une commande NMT doit être transmise pour retourner à l'état de fonctionnement. Le codeur supporte les messages d'alarme suivants.

Bit	Supported_alarms / messages d'alarme
0	défaut de position
1-11	
12	défaut E2prom
13...15	

Messages d'alarme (objet 0x6506/0x6505)

Bit	Supported_warnings / messages d'avertissement
0	
1	contrôle d'éclairage
2	chien de garde
3...15	

Messages d'avertissement (objet 0x6504/0x6503)

7 Objets spécifiques au fabricant

FR

7.1 Objet 0x5003, type de vitesse

L'objet 0x5003 met le temps de rafraîchissement et la résolution (points / seconde ou tr / min) de la vitesse d'information. L'objet influence l'objet de vitesse (0x6030) ainsi que l'objet d'accélération (0x6040).

Type de vitesse	Réglage
0	temps de rafraîchissement 200 ms, points / s
1	temps de rafraîchissement 10 ms, points / 10ms
2	temps de rafraîchissement 100 ms, points / 100 ms
3	temps de rafraîchissement 200 ms, tr / min

L'objet de vitesse est limité à une valeur de 16 bits avec signe. Pour éviter un débordement de données et pour optimiser l'exactitude, il est recommandé de calculer le réglage optimal du type de vitesse. Adopter également la mise à l'échelle dans le codeur. Le volume de données est limité et un débordement est évité, parce que le calcul de la valeur de vitesse est basé sur la valeur mono-tour mise à l'échelle.

Si la vitesse de rotation de l'arbre est plus de 1000 tr/min et le type de vitesse 0 (points/s) il y a un débordement de données. Dans ce cas, une résolution plus haute est nécessaire, c'est-à-dire points / 100 ms.

L'exactitude de la mesure de vitesse dépend de la résolution sélectionnée. Les chiffres dans ce tableau devraient être considérés comme valeurs de référence.

Type de vitesse	Rotation de l'arbre
0	> 100 tr/min
1	> 1000 tr/min
2	> 1000 tr/min
3	> 100 tr/min

Le tableau montre à partir de quelle vitesse de l'arbre l'exactitude de la valeur mesurée dévie de moins de 1 %. En général, l'exactitude s'améliore indépendamment du type de vitesse sélectionné, plus la rotation de l'arbre est haute.

7.2 Objet 0x5A03, numéro de série 2

L'objet 0x5A03 est un objet spécifique au fabricant dans lequel le numéro série peut être lu.

8 Exemple configuration de codeur

Cet exemple montre la structure du codeur pour la transmission cyclique de la valeur de position.

1. Régler l'adresse physique (n° de nœud) du codeur avec les commutateurs d'adresse. Pour plus d'informations voir le chapitre 4.
2. S'assurer que le débit de transmission de votre réseau CANopen et le taux de transmission du codeur soient identiques. Pour plus d'informations concernant le réglage du débit de transmission du codeur voir le chapitre 4.4.
3. Activer le codeur.
4. Le codeur transmet un message boot sur l'identificateur de défaut d'urgence (ID = 128 + adresse du codeur). Le message n'a pas d'octets de données.
5. La prochaine étape est la configuration du codeur par le message SDO. Pour régler une transmission cyclique de la valeur de position avec une répétabilité de 10 ms, il est nécessaire de transmettre une demande SDO (ID = 1536 + adresse du codeur) à l'objet timer cyclique (objet 6200h) avec les données ci-dessous. La confirmation par le codeur se fait avec la réponse SDO (ID = 1408 + adresse du codeur).

Octet 0	Octet 1	Octet 2	Octet 3	Octet 4	Octet 5	Octet 6	Octet 7
0x22	0x00	0x62	0x00	0x0A	0x00	0x00	0x00

Demande SDO

6. Pour mettre en service le codeur, vous devez transmettre un message NMT " Start Remote Node ", ID = 0, deux bits de données avec le contenu suivant :

Octet 0	**Octet 1**
0x01	adresse du codeur (n° de nœud)

Message NMT " Start Remote Node "

7. Le codeur est maintenant à l'état de fonctionnement et le message de position (ID = 384 + adresse du codeur) est transmis avec une répétabilité de 10 ms. Si un défaut surgit, le codeur transmet un message Emergency (ID = 128 + adresse du codeur).

FR